IMPLEMENTING NEW TECHNOLOGY

- PROCESSES AND PROBLEMS

by

WONG NGAR YAN SANDY
黃雅茵

MBA PROJECT REPORT

Presented to

The Graduate School

In Partial Fulfilment

of the Requirements for the Degree of

MASTER OF BUSINESS ADMINISTRATION

THREE-YEAR MBA PROGRAMME

THE CHINESE UNIVERSITY OF HONG KONG

May 1998

APPROVAL

Name: Wong Ngar Yan Sandy

Degree: Master of Business Administration

Title of Project: Implementing New Technology
- Processes and Problems

David C. Ahlstrom

Prof. David Ahlstrom

Date Approved: May 7, 1998

ABSTRACT

This paper studies on different processes undergone in system development and implementation. A case on how a company implemented a new system is used to study the processes. The whole system development and implementation process is broken down in ten phases : information analysis, system design, development, testing, system installation, training, system launch, review, system maintenance, and operation management. Various approaches and techniques used in each phases and problems encountered are examined. The pros and cons of each approach and technique will be evaluated through its application in the case. Theories and evidence in technology implementation are employed and illustrated through the case. Factors affecting the success of implementation in the case will also be analyzed. Finally, problems encountered and recommendations for the processes are summarized to give a generalized picture for the development and implementation of information technology project.

TABLE OF CONTENTS

ABSTRACT **4**

TABLE OF CONTENTS **5**

Chapter

1. INTRODUCTION **6**

II. METHODOLOGY **8**

III. INFORMATION ANALYSIS **11**

DEFINITION OF THE ORGANIZATION'S OR BUSINESS APPLICATION'S OBJECTIVES ... 11
EXAMINING THE CURRENT SYSTEM 12
PROPOSING SYSTEM SOLUTIONS 15

IV. SYSTEM DESIGN **20**

V. SYSTEM DEVELOPMENT **25**

VI. TESTING **30**

VII. SYSTEM INSTALLATION **33**

PARALLEL RUN AND CONVERSION 35
PILOT RUN 36
CUTOVER AND CONVERSION 38
HARDWARE INSTALLATION 39

VIII. TRAINING **42**

IX. SYSTEM LAUNCH **44**

X. SYSTEM REVIEW **46**

XI. SYSTEM MAINTENANCE **47**

XII. OPERATION MANAGEMENT **48**

XIII. CONCLUSION **49**

APPENDIX **53**

BIBLIOGRAPHY **55**

CHAPTER 1

INTRODUCTION

In the case studied, the company had a host-computer based system for processing customers' requests. Since the old system was a centralized host-base system, customers needed to send their request forms to the company's processing centre and the company's operation staff needed to process the forms and input them into the host system. This batch process is time consuming and prone to error. In order to reduce the manual work and provide faster service to customers, the company intended to replace the old system with a new one using a technology introduced by an outside vendor. In the new system, some of the processing is offloaded from the company's centralized host system and manual work to new equipment installed in the company's outlets. Customers' requests are processed instantly by the new equipment and information is passed back automatically from the equipment to the company's host system. A new host system was required to capture and process this information because the old system was not capable of processing information passed from the equipment. The new host system was also required to provide functionality similar to the old one for inquiry and maintenance of information required for processing customers' requests.

Four parties were involved in the whole development and implementation process : the vendor, users, system operation staff and system developers. The vendor was the supplier of the equipment and technology. Users were responsible for defining the functionality to be built on the new system, overseeing the whole implementation process and the actual operation of the system. Operations personnel were responsible for the daily system operation of the system, such as running of jobs and system support. System developers were responsible for the development of the new host system, communication and co-ordination with the vendor. Since the company is an international company, the system development involved users from different countries where environment and requirements differ.

In this paper, the whole system development and implementation process is broken down into ten phases : the information analysis phase which defines the business objectives, examines the current system and proposes system solution; the system design phase which translates business specifications into actual computer and manual processes; the system development phase which develops the system; the testing phase which tests the system in various aspects; the system installation phase which installs the system into live environment; the training phase which trains users for the necessary skills to operate the system; the system launch phase which puts the system in use; the system review phase which involves feedback and redesign process; the system maintenance phase which deals with capacity planning, measurement and tuning; and the operation management phase which addresses productivity measures, workload projection and technology tracking. Activities, deliverables and problems encountered in each phase will be studied in the paper.

CHAPTER 2

METHODOLOGY AND INTERVIEW SAMPLE

Throughout the paper, various approaches and techniques employed in the development and implementation of the information technology project in the case are studied in order to give a generalized picture for technology implementation. Theories and their applications in the case are described. Analysis and recommendations are made on the activities undergone in each phase. Information was collected through interviews, participation and observation of activities undergone.

The aims of the interview are to find out how different people perceive the system, what their concerns are in system development and implementation, and what problems they encounter in the processes. Semi-structured interviews in the form of casual conversation were conducted with various parties, such as users, customers and system developers involved in order to provide an open atmosphere. The interviewees were generally asked how they feel about the system, what they expect from the system, and what problems they encounter in defining or using it. Different groups of people were interviewed to provide different angles on the whole implementation. It is found that people at management level had more concerns on meeting business objectives within their budget rather than the actual development and operation of the system. They had high expectation on the capability of the system. For example, they wanted to have a very flexible and secure system with fast processing speed. However, they required the system development lead time to be as short as possible and the development cost to be low. Operation users were very concerned about the actual running of the system. They focused more on friendliness of the system, ease-of-use, processing speed, and robustness of the system. For instance, some operation users complained that too many keystrokes were required to perform a single transaction using the new equipment, response of the new system was too slow, and too many screens of input were required to process information in the host computer system. System developers had more concerns on the feasibility of the system proposed as they were responsible for the development and maintenance of the system. Due to their technical background, they tended to concentrate on technical issues and pay less attention to human aspects such as users' business needs or actual operation of the system. Customers were concerned about the speed and correctness of the system. Several problems were encountered with gathering the information, for example, some interviewees were too busy to be interviewed, some were not willing to express their feelings, or some were not able to express their ideas clearly and effectively. However, the semi-structured type of interview helped them to express their ideas more freely.

Besides interviews, information was also collected through observation and participation in the processes. The effectiveness of various techniques, such as user group session and prototyping, employed in system development was evaluation by observing the users' response. Application of different techniques under different situations was also studied by observing how people chose amongst these techniques in each scenario. The response of users towards using the new equipment and new system was observed to give hints on how to make the implementation of new technology more acceptable to users. Human issues such as conflicts and communication between different groups, morale of people involved in the technology implementation processes were examined by observing the interaction between these groups and the attitude of the people. Problems in different phases in technology implementation were studied through observing the activities undergone.

The process of examining the current system was studied through participating in the review of existing documentation and interviews with users. Techniques used in system development were evaluated through participation in the actual system development. The approaches used in system installation and training were analyzed using the information gathered in participating the actual processes. The problems encountered in each phase were revealed and sound recommendations were then made based on the information gathered through participation.

CHAPTER 3

INFORMATION ANALYSIS

Definition of the Organization's or Business Application's Objectives

This refers to the definition of business specification which states how the user's business needs should be met by the proposed system. This defines the scope of the system, determines who is affected by the system, and makes rough estimates on benefits and costs of the project. Both systems reasons, which may include limitation of processing capabilities, vendor support, etc., and business reasons for the project need to be stated. A general statement of project cost is also needed for cost benefit analysis. Various subprocesses can be undertaken to help cost determination. These are perusing and assessing existing documentation; roughly stating desired system results, noting hardware, software, and vendor constraints; generating a rough draft of potential alternatives with accompanying costs for hardware, software.

In the case studied, the application of the new technology was initiated by the vendor in some technical conferences held regularly for companies within same industry. The objectives of the system are to reduce the manual work required and provide better service to the company's customers. Although the application's objectives were clearly defined, not much had been done on cost benefit analysis. Both the development effort and market response had not been estimated. Moreover, since the idea was suggested by the vendor, not much had been done to search for other potential alternatives to serve the objectives. The company should have a better understanding on the technology and its application before engaging in it. It may also explore some other alternatives in order to make the best choice.

Examining the Current System

The main objectives of this process is to understand and document the structure and purpose of the current system, which can be totally manual, partially or fully automated, in order to determine the requirements of the new system so that the new system can fit into existing practice and procedures of users and customers without disruption. This process can be further broken down into several subprocesses, such as converting information of data flows and procedures of current system into logical descriptions; gathering and analyzing data in old system; and determining data and processes required in the new system. The usual techniques employed are interviewing, questionnairing, peer/user reviewing, organizational modeling, and observing or participating in user's actual work.

In the case studied, the existing host system was to be replaced by a new one. In the initial phase of the project, system specifications of the old system were studied so that the user interfaces and processing of the new system can imitate the old one, making it easier for the users and customers to adapt. However, since the specification of the old system is not comprehensive enough and enhanced features had not been properly documented, some features were missed out in the initial design of the system. The need to re-establish the required features into the new system causes discrepancies in original effort estimation, which in turn resulted in unexpected overtime work of system developers. This degraded the quality of the new system and affected the morale of the system developers. This avoidable re-do processes also made the system development ineffective and unsystematic.

Besides the incomplete documentation, system developers also faced problems in gathering information from users. Since no documentation of procedures was available, system developers relied heavily on person-to-person communication to understand how the old system runs. However, the users were all overloaded with daily operation work, thus it was almost impossible for the system developers to interview them to get information on their existing work practices and requirements. The division of labour was another obstacle in examining the current system as the users' operation work is broken down into very small sections such that nobody has an overall picture while system developers just did not know whom to interview. This is a common problem with complex systems (Perrow, C. 1986 Complex Organization).

Although there is a system coordinator held responsible for assisting system developers in development, he was newly employed and was not acquainted with the system or the company's operation, making his work relatively ineffective and inefficient. To make things worse, he was also responsible for all system related matters, thus he was fully occupied by daily operation work and did not have time to communicate with the system developers. Also, since he was not given the authority to collect the information from users, some users were reluctant to provide information to him, especially when they were busy with their own work. Therefore, the system coordinator was unable to provide much information.

The company also needed to emphasize the importance of communication between its users and system developers. A system coordination team, instead of a single person, should be set up to handle daily operation matters as well as assist system development. This team should be delegated the authority to collect the required information. Time and effort should be budgeted for users to providing information for system development. Besides, organization modelling should be constructed so that system developers has a clear picture on the roles of different users so that they know whom to interview for system analysis and whom to inform when system changes are to be performed.

Besides, documentation should be done more properly. A complete set of documents including functional specification, technical specification, test plan, manual and user procedure should be kept systematically in some documentation libraries and made available to both users and system developers. Moreover, there should be some standards on the documents to ensure they are comprehensive and complete. Defining standards in writing is helpful as it guarantees consistency and reduces costs of maintenance. Documentation should be made as part of development work with time and effort budgeted. The importance of documentation should be emphasized across all management levels and compilation of documentation should be made compulsory.

A documentation database can be built to contain data about who is concerned with the system, what data and processes are part of the system, why the system is needed and how the system processes the required information. Documentation should contain list of input and output to the system, business functions included within the system boundary, data connections between business functions. A comprehensive data dictionary or data encyclopedia can also be constructed. This can lower the cost of maintenance and restart, reduce the need for verbal communication, and provide idea of progress when it is built during the development of the system as the progress can be reflected by the information available in the database. In the case studied, although there is no formal documentation database, system developers compiled some documents on various system aspects such as processing logic, input and output information required. However, the documentation is incomplete as only system information exists and not much human issues, such as procedure flow, of the system is documented. This makes future maintenance and enhancement difficult as human factors may be ignored in design.

Proposing System Solutions

In this process, the boundary and functionality of the system is defined. Unless the new system is intended to introduce an unprecedented innovation, systems that other organizations have already developed and standard software solutions can be a useful source of ideas for requirements. Different approaches to user involvement can be employed to aid user and system developer to define the system, such as an experimental approach with the use of prototype; and a participative approach in which user group sessions are conducted with technical personnel.

Prototype is a small or "scale-down" version of the system developed with its most pertinent characteristics. During prototyping, the prototype is developed and presented to users for comment. There are two types of prototypes : explanatory prototype which is used to clarify some ideas or some of the systems' features to users; experimental prototype which is used to test new aspects of system. Prototyping provides better communication channels between users and system developers, this in turn creates the harmony which promotes creative solutions. It helps users to visualize the new system's features and refine their requirements as the users can actually try and use the prototype. The constant user interaction involved in prototyping places the user in the role of designer and the system developer in the role of builder. Users also acquire confidence and become better prepared for the changes brought by the new system through the demonstration of the prototype. Technical problems arises during prototype building can warn of potential technical failures.

Besides, by using prototypes, alternative designs can be tested before committing to a single development route. However, the effort and costs required to build the prototype can be prohibitive. Prototype may also provide inaccurate performance standard as compared to the real system. Also, users must be willing to commit time and energy to work with the system developers in prototyping. In general, the use of prototypes is more beneficial when the degree of user experience with the system development is low that they cannot express their requirements precisely, or when the features of the new system are not well understood.

Both experimental and participative approaches were used to define the system in the case studied. Since the new system would be launched in other countries, user group sessions were conducted for users from all involved countries. System developers conducted "system walkthroughs", giving them a brief understanding of the system and also the chance to give feedback to system developers. Users from different countries also discussed their requirements and resolved conflicts arisen from difference in business environment. These sessions also served as brainstorming sessions for the users to explore various possibilities of the new system and helped to define system functionality and boundaries.

Since the new host system was mainly aimed at replacing the old one with most of the functionality based on the old one, many of the features are already well understood. Moreover, there was not enough time to build a large prototype for the whole system. Thus only a small explanatory prototype with the totally new features was built. A prototype of the new equipment was also demonstrated to the users. This was found to be much more efficient and effective than verbal or written communication, and, it was more readily accepted by users, although there was a cost for building and demonstrating the prototypes.

However, the personnel involved in these sessions were quite senior, they could only give high level suggestions. Some of them were not too concerned with the actual operation of the system. Moreover, some of their "blue sky" requirements were found to be unattainable, both in a technical and operational sense. Some requirements were also found to be incomprehensible to the actual users when the proposed system was presented to them. Important issues such as ease-of-use and efficiency had not been considered by the senior users. Some senior personnel did not consider the operation environment, education level and background of the actual users, making the system they proposed far too complicated to operate at user sites. This required certain subsystems redesigned and caused wastage in system development effort.

Besides, users submitting the requirements were inexperienced in system development such that their requirements were incomplete and misleading. Some requirements failed to stipulate operational schedules, target operating costs, or performance norms. This caused part of the system being eventually rejected by the actual users who work under the system and caused rework. This disruption and unnecessary rework could be avoided if the actual users were involved in system definition. Moreover, there were unrealistic expectations and system developers needed to make value judgements between dreams, wishes, desires, and necessities. Overview, standard procedures and guidelines can be used to help users to specify their requirements.

Since the users and system developers are generally under different units with different working environment and management, it is very easy to fall in the trap of a "We-They" relationship during system definition, especially when something goes wrong. In the case, certain "We-They" syndrome was exhibited when users ignored the constraints in system development and system developers disregarded the business needs of users. Since users had the ownership of the project, they tended to impose their requirements by exercising their authority. This prohibited open dialog between system developers and users and hampered the cooperation and communication between these groups, which in turn affected the morale of the system developers. Better mutual understandings between users and system developers should be promoted so that each side can be aware the other side's concerns and limitation. Steps such as establishing a joint participation cooperative base can be used to reduce the "We-They" syndrome.

CHAPTER 4

SYSTEM DESIGN

This is the translation of the specifications of business into actual computer and operator processes. The deliverables in this process are functional specifications, which describes the desired output to be produced, handling requirements that defines the manner in which the system is to be embedded in the work and communication process of the users. User involvement has been crucial in this stage as they provide information for tuning the system to fit their needs.

Feasibility assessments on legal and regulatory feasibility, organizational feasibility, social feasibility, technical feasibility, economic feasibility and operational feasibility are required. Legal and regulatory feasibility refers to any external imposed requirements or conditions that an information system must meet. Organizational feasibility refers to the changes that a proposed information system entails for the structure of an organization, its management procedures, and the way decisions are made. This also examines whether a proposed system affects the integrity of the organization's functioning. Social feasibility refers to the acceptance of changes in work conditions, power structure and relations by those who will work under the system; and also the change in the relations between the organization and its customers. Economic feasibility refers to consideration of costs of implementing the system, continuing the systems development, preparing persons and machines for the system, and maintaining the system over its life. An operational feasibility assessment ensures the system will be accepted without destructive resistance. Besides these feasibility assessments, manual work process, such as movement of paper and material in the environment where the system is installed, should also be considered in system design.

In the case studied, legal and regulatory feasibility had not been considered in which an important information requirement imposed by external party had been left out. This was afterwards redesigned, but effort had already been wasted. Not much was done in assessing organizational feasibility and social feasibility as the new system was just a replacement of the old one. Thus operational feasibility was not fully assessed in which user's work flow had not been carefully studied during system design. Some functions proved far too complicated to use or they required information which was not available to the corresponding users. This led to redesign of certain functions and re-engineering of some of the user's work flow after the system was launched. This had caused unnecessary disruption to both system developers and users.

For technical feasibility, vendors were involved to ensure the system was feasible. Although the new system is a replacement of the old one with many features retained, many new features were made available with the introduction of new technology. Users explored various new product features, system developers translated these requirements into technical specification. Afterwards, system developers evaluated the feasibility of incorporating such features in the new host system and the new equipment and reviewed this with the vendor. When the proposed features were not feasible, system developers communicated the technical constraints to users and then users redesigned the system features. Then system developers assessed the new requirements again with vendor for feasibility analysis. This process was iterated and finally generated system features which were feasible and satisfactory to the users.

During the system design, system developers revised vendor specifications with vendor. Since the system developers did not possess the required technical knowledge, the vendor specifications were written mostly by the vendor. This made the specifications too technical and focused only on the actual processing of the vendor's equipment and the associated software rather than the requirements of the company. This causes difficulties when the company wanted to find other vendors to perform the same task as it could not specify clearly its requirements to them. It was only after the company had faced problems with other vendors the system developers added in the company's requirements into the vendor's part of system specification. If time and effort were budgeted for the system developers to review the system specification, the document can be more accurate and useful.

Moreover, since the technology was totally new, both system developers and vendors faced lots of problems in compiling the specifications. The system specification was incomplete and subjected to change during the whole development process. Some changes were not caught up in the specification and were left out in subsequent development. However, it is inevitable as the technology was relatively new to the company and the application was complex to the vendors.

Besides consideration on feasibility, the system should be designed in a humanized and logical fashion such that the information conveyed is sound and useful. System developers should know their users, can be temperamental, trained, naive, computer-literate or illiterate, computer lovers or haters, etc. System should be designed with consideration of the needs and capabilities of users. For example, efficiency will be crucial in an environment that has low tolerance for frustration, or, a system needs to be fault tolerant in an environment with naive users.

Old systems should not be ignored when designing the new system because even though the new system may have been better overall, slight deficiencies may have detracted from its appeal and slowed its acceptance by users. Functions supported by the old system should be considered necessities since the user's organization is structured around the functions now available. Functional equivalence should be preserved in the design so that similar or related function will have similar user interface, and the changeover will not be perceived as too drastic. In this case, some functionality in the old system was not incorporated in the new system. This caused discontent and surprises from users when they used the new system. If these functionalities were maintained in the new system from the very beginning, it will be more readily welcomed by the users.

In system design, tradeoffs between cost of protecting and cost of loss, and between ease of use and integrity need to be considered. Integrity is established by adding redundancy. Improve integrity can reduce system efficiency by introducing system access barriers, increasing complexity of operation, and forcing extra computational activities. Similarly greater integrity can make system usage more difficult. Safety and risk reduction can also degrade performance.

Functional specification, the major deliverable in this phase, must provide user with a clear view of the outcome of the systems product, the changes to the user organization, changes in business process in accommodation of the changed system, risks of the application, and steps to be taken to make the system work. User participation will be helpful to ensure the functional specification can achieve the above. However, in the case, the functional specification was prepared solely by the system developers such that it concentrated mostly on how to operate the system and details on input/output formats, whereas business and operation issues were omitted. This deprives the usefulness of the functional specification in analysis for future enhancement.

CHAPTER 5

SYSTEM DEVELOPMENT

During this phase, the system is actually developed. User procedures and operational documentation are compiled. Two basic methods of constructing the system -- the bottom-up and the top-down or versioned approaches -- can be used. In the bottom-up approach, system is accomplished by building and testing all programs and procedure subsystems independently. Subsystems are then integrated and tested in combination. The difficulty with this approach is that the subsystems may not always fit together as planned with interface problems arisen, resulting in rework of the subsystems at the interface boundaries. This causes rippling effects within the subsystems, which may cause problems throughout the whole system architecture.

In the top-down or versioned approach, the basic structure of the system is built first, providing developers with an opportunity to establish and test interfaces between subsystems. Then, detail is added to the structure, resulting in more complete versions of the system. The advantages of using this versioned approach are the incremental learning users get by participating immediately in the use of the system. This helps uncover any weaknesses in the system and gain acceptance through communication. User confidence will grow and also disruption to the user will be minimized as system is launched incrementally, rather than as one big push. However, there are some disadvantages in that users have to be briefed clearly that the system is not really complete until all subsystems are completed and they may need to re-adjust their operation when incremental changes are applied at subsequent stages. Another disadvantage is that the hardware and software have to be in place for early versions where these may not be available or require the capital which is not providing immediate value.

In the case, bottom-up approach was used because the new equipment was available only at a late stage of the whole system development. Since the whole system was very large and composed of several subsystems, it was designed and developed in parts by different sub-teams. However, due to the tight development schedule, only a few meetings were held with developers of the subsystems to ensure system integrity and compatibility between subsystems. Thus there were occasionally some misunderstanding and miscommunication when designing interface between subsystems. This caused rework of the subsystems and increased workload of system developers.

During the whole system development stage, users changed the requirements frequently. This increased the workload of system developers because once the system is built, it is not easy to change. The launch date of the project was also delayed because of the difficulties in using the new technology and extra effort required to handle the change in requirements. This also made the compilation of documentation such as functional specifications, technical specifications and user manuals difficult, partly because corresponding parties such as some developers were not well informed about the changes in requirements, or they were just being too busy to keep up with the changes. It also made planning and effort estimation inaccurate. The requirements should be frozen at a certain point, say, after the functional specification is approved. Alternatively, the extra user requirements should be deferred to next phase of the project so that the project can be launched in time. This phased launch of the system also helps users to refine their requirements and makes them more realistic in defining their requirements, as they can have a better picture of the system in live environment after it is launched.

The developers should conduct some briefing sessions to discuss the features of the design so that other developers will have an overall picture of the system. However, in the case, the developers were too busy to conduct any review session with other developers. This deprives the chance for sharing knowledge on the system and skills, as well as the chance to find errors and omissions. This made the whole project rely heavily on several key developers and caused an uneven distribution of workload. This may also result in bias in opinion because of personal preference and the person's familiarity with existing equipment. If the review session could be conducted, the new technology can be shared among the whole team, making it more capable of maintenance work of the system. This also provides visibility, highlighting and detailing of the project contents, deliverables and outcomes to all involved. Moreover, this creates wider awareness of the system and formally transfer ownership of a system from few developers to the whole project team, making the development effort less personal and more a group effort.

As the technology employed in the case is totally new to all system developers and no training was given on the necessary skill, developers had to train on-the-job using their own time and this affected staff morale. Plans should include training if skills required is not available.

In the development phase, since many system developers involved were inexperienced in the whole environment and the experienced developers were too busy with their work, they were often ignored and not fully utilized. To make better use of resources, cost of learning should be budgeted so that the experienced developers could have time to mentor newcomers. Also, pairing can be created so that newcomers have a better understanding on their work and in turn gain job satisfaction. Moreover, this will promote sharing of skills and create logical source of backup for the key developers.

Besides, the assignment of tasks had not been clearly specified to each individual to ensure all know their responsibilities in the project. This led to lack of commitments for work performance and created confusion, resulting in some subsystems actually being left out in development. Although the subsystems were eventually developed, it was at the expense of some highly motivated system developers. This created some ill will between different developers that could have been avoided if the tasks were assigned clearly.

In addition to communication within the team, cross team communication is also necessary in the case studied since the company has system developers stationed in different countries to act as local support. System developers from other countries came to Hong Kong to understand the system as well as to share the knowledge. This is found to be very effective as they got hands-on experience in the system. They then became capable of supporting the system in local countries as well as propagating the knowledge to other local system developers.

In the case, there was no quality control throughout the development process. Some of the subsystems were poorly developed and this resulted in high costs in maintenance and future enhancement. To improve the situation, there should be some control on quality of the delivered results and consistency to standards. Quality should be addressed continuously in the project and the developers should be trained in good quality techniques. Regular review meetings should be conducted to set metrics for measuring quality, to revisit all major areas of quality in the project, and to follow-up on all quality flaws and promote quality results.

CHAPTER 6

TESTING

In the testing phase, the system is tested to ensure it is reliable, operative and fulfills user requirements. There are three basic types of testing, namely unit test, which tests modules and subsystems independently, integration test which tests the combined subsystems, and user acceptance test which ensures the system is acceptable to user. Several tests are also needed to uncover system weaknesses. They are : stress test which presents the system with large quantities of data over a specified time period; the volume test which presents the system with a peak load of data; the usability test which presents the user with the system to detect unfriendliness; the security test to prove the system does not have integrity problem; the performance test to ensure response and processing time are at acceptable level; the documentation testing to ensure the documentation is adequate and accurate; and the procedure test to ensure the procedure allows user to achieve required performance. Performance may be tested using benchmark tests, where prespecified performance is compared with actual performance.

In the case studied, both unit test and integration test were performed. However, due to the tight schedule, the system was still under development when the testing was done, making the testing result inaccurate.

For the user acceptance test, since user were too busy to write test plans, the test plans were incomplete, or even never accomplished. And, the users who constructed the test plans had not been involved in defining the system, they just did not know the functionality of the system and thus did not know how to test it. Therefore, system developers explained the system to them and helped them to make up the test cases. This defeated the whole purpose of user acceptance test because system developers constructed test cases according to their own understanding on the system which may differ from those of the users. Heavy involvement of system developers in test case construction also biased the testing result. The test cases were found to be incomplete as only valid values and valid combinations were tested whereas invalid input had not been tested.

Also, since the users were not acquainted with the system, they did not know what was expected and could not report any system malfunction, although there are standardized procedures and documents for reporting errors and users were well aware of that. The testing eventually became a session for sharing system knowledge with end users during the hands-on experience gained from testing instead of actual testing of the system. However, the testing sessions gave information on how the users perceive and use the system and generated feedback on the system which is very useful to system developers for refining the system.

Besides problems with test plans, users were not available to perform the tests. The users did not have clear responsibility to perform testing for the system and they were fully occupied by their daily operation work. Therefore, they could not put much effort on the testing, making the testing incomplete. User departments should budget resources for testing and make system testing part of job responsibility of their staff.

There had been no testing of both documentation and procedure. This made the correctness of the documentation and procedure rely heavily on the persons who wrote the documentation and procedure. This created a heavy workload for them and is characteristic of an organization that learns very slowly; through trial and error.

CHAPTER 7

SYSTEM INSTALLATION

There are several approaches to systems installation, namely parallel, phased, pilot, and cutover approaches. However, these approaches are not mutually exclusive and may be combined to build the entire system. The parallel approach is a safe way to check out the system. System activities are duplicated as the old system and the new system are both operated simultaneously. The new system is compared with the old system during the parallel run period, later the old system is abandoned and the new one goes on only after it is proven to run correctly. However, the parallel approach may not always be possible as the new system may be incompatible with the old system, or there may not be room to house both the old and new system. Also, cost of running duplicate facilities can be prohibitive. Parallel approaches may also take the longest time as people may put less energy on the new system as they are expending energy running the old one. Moreover, the verification of the new system by comparing the results of the new system against those of the old system relies on the correctness of the old one. Parallel implementation also has a real dilemma of whether to assign acquired temporary personnel to aid in the conversion to run the old system, which they know little about, or to assign them to the new system, precluding "regular" workers from the new experience. Nevertheless, parallel approach is less stressful as there is something to fall back on.

With phased approach, different functional parts of a system are brought to life one after another to add the system features in a function-by-function manner. This is specially useful when the users are unwilling or unable to take up the changes brought by the new system, or, when the requirements imply a drastic change in the way the users conduct their business.

If a major change in user operations is dictated by the goals and objectives set by management, system developers should be aware of the system being too complex or the rate of change being too great for the users to assimilate. This can be solved by phased approach where the system is installed incrementally, with the simplified version installed first and additional increments of function installed in manageable steps only after the users have become comfortable. This reduces the risk of impairing the functioning of the organization and provides opportunity for gradually learning the new system. However, this requires breaking the system into various functions and it is not always possible to factor a system into meaningful subsystems. It may also break down if there are complex interfaces between activated subsystems. It is feasible only when system functions can be isolated and results can be adequately confirmed according to system test criteria.

The pilot approach requires the installation of the new system in sites that are representative of the complete system. The new system is installed in full only after it is proven to be operable at the pilot sites. Meaningful selection of pilot sites is essential for pilot approach. There may be no representative sites, or it may be politically unwise to make such choice. It also may be impossible to decouple one site in a dependent network. Pilot approach generates stress on the pilot sites and it may produce misconception on the system if the pilot sites are not correctly chosen.

With cutover approach, old system is changed into new system overnight. This provides no turning back and no hidden noncommitment. The cutover approach is most stressful as there is no turning back. However it provides cheap and fast means of implementation, and, it is time and cost saving.

In the case studied, since the whole implementation involved operating a new host system which replaced the old system and using new equipment in outlets, parallel approach, pilot approach and cutover approach were used. Parallel approach was used to test the operation of the new host system in live environment while pilot approach was used to test the new equipment in selected outlets. Finally the old system was cutover to the new one. As the new system was to replace the old one where information in the old system needed to be carried forward to the new one, conversion was performed. Conversion was done twice, one before the parallel run to provide live data for the new system. After the new system was proven to run correctly, conversion was performed again to bring the most updated information to the new system and the old system was then cutover to the new one thereafter. This hybrid approach enabled the system to be tested extensively in a live environment.

Parallel Run and Conversion

In the case studied, since the new system is to replace the old one, parallel run approach was used to ensure smooth transition to the new one. A trial conversion of the old system was planned, then both the old and new systems run in parallel with results compared. However, there had been system constraint on storage space for the converted data, thus the conversion for parallel run was delayed. Due to the tight development schedule, the data in the old system had not been checked before the conversion for parallel run. The incorrect data had been brought across to the new system. However, since the users were too busy with their daily operation work, the checking on the data in the new system was incomplete. Although system developers had developed some automated checking to compare the results of the old and new systems, the difference in processing logic between the old and new systems made it impossible to compare the results. Moreover, the users were just too busy to verify the results of the new system. This made the parallel run ineffective.

Moreover, since the old system was still in operation, the user just did not bother to try the new one, especially when it doubled their effort to update information in both systems. This made the comparison of data in old and new systems impossible. Despite all the problems, the parallel run nevertheless gave hands-on experience to users in using the new system which built up their confidence and ensured the new system is operable in production environment.

Pilot Run

Since the system implementation involved installation of new equipment in the company's outlets, a pilot run was conducted to ensure the new equipment function in live environment and to give users hands-on experience in using it. The advantage of pilot run is that the system can be fully tested in a live situation before it is adopted. Pilot run can also locate problems that will not occur in laboratory environment which is much more stable than live environment. Various operation related problems were uncovered during pilot run, giving good feedback on the system. This provides useful information for future development and enhancement.

In the case, the new equipment was installed at selected outlets. Since the system had not been launched publicly during the pilot run, it could only be tested by those involved in the project, namely the company's staff. Test cases were assigned and questionnaires were given to staff involved in the pilot test. A briefing session was also conducted to all staff involved to ensure they know what to do. Incentives such as gifts given for returning questionnaires were used to encourage them to do the testing. Since most involved in the test were involved in the whole system development, they were very cooperative and the pilot test was successful as it spotted several problems such as problems in hardware, software, and operation procedures. These problems were then consolidated and fixed.

The pilot test also uncovered some serious hardware problems which never occurred during testing because of the difference in environment. Although this took a long time and lots of effort to fix and delayed the launch date, this saved the whole project as it was discovered and fixed before the project was fully launched.

Besides testing the system, the pilot run gave information on how the working level users perceived and used the new equipment. For example, some users at pilot outlets found the equipment too difficult to use, and the combination of keystrokes too confusing. This was helpful in refining the equipment and operation procedures before the system was fully launched. Various operational issues encountered in pilot sites gave hints for the preparation of training materials. Moreover, the pilot run helped to train the users at pilot outlets because those performing the test are acquainted with system, they passed this knowledge to the users at pilot outlets.

Cutover and Conversion

In this process the existing files and databases are converted into a form usable with the new system. Specification is required to collect and establish the required databases which currently do not exist. Such data may come from outside sources or from existing data. Care must be taken to ensure the validity and existence of data in current system. Spot checks of actual data files might be done. Trial run of conversion may also be performed to ensure smooth conversion. Checkpoints should be made to verify the correctness of the converted data. Documents and memos should be distributed to all those involved in the conversion to ensure everyone understand their roles in the conversion process.

In the case, the conversion was done with detailed planning. The schedule was designed with consideration of possible disruption to both the system and users' operation. The schedule had been circulated to everyone involved. System developers conducted briefing sessions to ensure all involved understand the whole process. Arrangement had been made to have both users and developers to stand-by and check the data right after the conversion to ensure its correctness. Automated integrity checking was also developed to help user to check the conversion result.

However, some unexpected errors in the data in the old system were brought across to the new one and this required extra effort to fix. Nevertheless, the system was cutover to the new one. The functions in the old system were ceased to operate and the users were forced to use the new one. Although there were some operations issue during the transition period, the hands-on experience the users had gained during the parallel run period alleviated the disruption.

Hardware Installation

In this process, installation plan is essential and it should include the following aspects : site specifications which includes air conditioning, storage space for supplies, and security; modifications to existing hardware and systems software; location sequences if multiple sites are included; installation sequences when different components are required; and training sequences to assure new hardware and software can be operated and maintained. Also, the installation plan should be aimed at minimizing the disruption to users for which the system is installed.

In the case studied, the new equipment was installed at various outlets. Installation schedule was worked out between the company, vendor and users at the outlets. Location sequence was an important consideration as there are outlets at different areas and the most efficient way to install the hardware became crucial. Besides, some locations were selected to be installed first because of their high transaction volume which could be used as a stress test on the system.

Issues like disruption to operation at the outlets, special events occurred at the outlets such as festival, renovation and open day, and requirement by different user groups were considered. For example, hardware installation was avoided during festivals or open day when outlets users were busy and disruption was not welcomed. Outlets with several sets of equipment required the hardware to be installed one at a time so that they could have the old one as backup in case the new one failed. However, this implies the installation team needed to visit the same site several times and this lengthened the duration required for installation.

Besides the scheduling of installation, the company also faced problems in the actual installation. The vendor had under-estimated the time required to install each piece of equipment. This made the installation schedule very tight and caused unexpected overtime work of the vendor's installation team. However, the unanticipated overtime work arrangement had not been made and this caused dissatisfaction on both the company and the vendor. Moreover, the installation teams were not that well-trained to install the equipment and this caused errors. Although the vendor eventually wrote an installation guideline for the teams to follow, disruption had already been made and effort was required to correct the errors. Moreover, since the teams were not that acquainted with the equipment, some of them were not able to demonstrate the correct operation to the end users. Some even failed to operate the equipment and this gave a bad impression of the equipment to the end users. Also, some failures had not be reported and later on found inoperative at the outlets.

To prevent this from happening, a better estimation should be obtained from vendors. Also, installation guide should be prepared and the installation teams should be trained before the actual work. Incident log sheets should be distributed to the installation teams for them to report any problem occurred so that they can be fixed more promptly.

CHAPTER 8

TRAINING

The purpose of training is to ensure users do not exhibit any dysfunctional behaviour and system resistance. Moreover, training is intended to provide skills and awareness to a variety of categories of people who will be a part of a new system. For certain systems, training must also be provided to people outside the organization. There will be need to communicate to the outsiders and also a need to train people to deal with enquiries. Training also ensures people are ready to perform once the system is put in place. To have successful training, tasks the target population must do and desired performance on the tasks must be defined. Training must be centered around tangible things like forms, and manuals. Trainees must respond actively and receive timely feedback. Timing is also important as there is a difficult decision on whether to train people early and risk their forgetting or the system changing, or to train people close to actual use and not allowing time for practice or absorbing the material.

To assist training, a training library and operations library may be created. Training library may contain training manuals, presentation materials, forms and other samples, reference manuals, video/audio tapes and materials, training computer software, on-line help software and training guides.

In the case studied, although there are documents like manuals, forms and sample, reference manual, the documents are scattered since there is no training library. This makes it very hard for new users to pick up operation skill as they do not know where to find the documents for reference. This in turn made the training rely heavily on verbal communication. Also, since the manual was prepared by system developers, operational issues were omitted and this reduces its usefulness.

Since the vendor delivered the equipment at a late stage, training was conducted very close to system launch. Many users found that they did not have enough practice before the actual use. Besides, the new equipment was too complicated to use, making them reluctant to use it. To remedy the situation, cue cards were designed by the vendor and distributed to end users to assist them to use the equipment. However, the cue cards were too brief and incomprehensible. It was also technical and not user-friendly. Afterwards end users were then interviewed, and the cue card was redesigned and distributed.

Moreover, since there were too many people to train, the company took a train-the-trainer approach. Training sessions were conducted by system developers and vendor for the trainers, then the trainers passed this knowledge to the lower level operation staff. There were some advantages of this approach, such as saving training costs and the trainers can tailor their training materials to the specific operation environment of their subordinates which is unknown to the system developers and vendor. However, there were some disadvantages as some trainers were quite senior and they did not understand the actual operation flow, or, they were not willing to learn to use the equipment. Some trainers did not understand the operation procedures of the equipment and passed some wrong information to their subordinates.

CHAPTER 9

SYSTEM LAUNCH

In this phase, system implementation plan is prepared and includes coordination of the available hardware, software, and user/operations procedures. It should specify what must be done, by whom, and which system objectives are to be met. Representative users and operations personnel who will participate and review the materials produced must be involved. The plan should be reviewed and approved by system developers, operation staff, business user, and technical support personnel to prevent any surprises and ambiguities in roles. Resources requirement of the implementation must be estimated.

In the case studied, system developers prepared the implementation plan to assist the launch of the system. The plan includes system work to be performed, parameters to be input by user and results to be checked. Meetings had been held to ensure the schedule of the plan was feasible and to ensure all involved parties understand their roles. This is found to be effective and it smoothened the system launch.

As the new system has many functionality imitated the old one, users did not have big problem in adapting to the new one. Moreover, the hands-on experience they had gained in parallel run period alleviated disruption on the first day operation of the new system. However, user still faced problems when they first used the system as some of the processing is different from the old one. Gap analysis was then written to explain the difference and analog the new system to the old one to help user to understand the new system. If the gap analysis had been done earlier, the user would have faced less disruption and anxiety with the new system.

CHAPTER 10

SYSTEM REVIEW

This phase involves the feedback and redesign processes. During system review, the system is examined to determine whether user requirements are being met and to tune the system to improve processing efficiency. The review should consist of the evaluation of the system development efforts such as development approach and techniques used; evaluation of the developed system such as overall performance, usability, maintainability, dependability. The review should also include non-intended impact of the system on the organization's functioning and performance.

In the case, post-implementation review sessions were conducted to review the result. However, there were only sessions held by individual party instead of a cross-departmental session. This made the review sessions only focused on local areas. Also, the reviews only focused on problems encountered and identification of the parties responsible for the problems rather than preventive or remedial actions. They did not include development approach and techniques. This made the review not constructive and incomplete.

CHAPTER 11

SYSTEM MAINTENANCE

This phase covers capacity planning, program extensions, program improvements, measurement and tuning, and system reconfiguration. This is an ongoing process for maintaining the system. Capacity planning includes the planning for increase or decrease in storage space or processing capacity due to change in business volume. Program extensions addresses the incremental changes not catered in initial design.

In the case studied, not much had been done on capacity planning. System developers were allotted other projects after the system was launched. So they did not have time for ongoing plan of the system. Capacity issues were dealt with only after the loading reached a dangerous level. Although there were measurement and tuning exercises for the system, there was not enough resources allocated. In addition, the effort to tune to system to achieve the required service levels in processing speed was overwhelmed by the growth in business volume.

To improve the situation, system developers should be informed about the projection to growth in business volume so that capacity planning can be more realistic in system design. Resources should be budgeted for ongoing maintenance and refinement of the system after it goes live.

CHAPTER 12

OPERATION MANAGEMENT

This phase addresses the system operation, service measures, productivity measures, workload projection, saturation prediction, analysis of improvements, justification of upgrades, technology tracking and continued training. This includes system availability, processing speed, efficiency. Operation library may be built for operations management which includes job run sequence, job control command, job restart instructions, hardware and software resources, run commands, normal action sequences, application recovery procedures, and contact person. Statistics on job run time, response time, disk space utilization and file size can be collected for operation management.

In the case studied, there is a well-developed operation library as the company has standards to enforce documents like job run sequence, restart instructions to be well-written. There are documents on the description of various jobs, how to run them and how to recover if they fail. System developers also conducted briefing sessions to instruct system operators to run the jobs. The company's system operation staff also has tools for system health check and to monitor any possible system overload. However, since the system operation staff do not have information on business volume growth, they cannot make good workload projection. This affects the timeliness of actions to cater change in workload.

CHAPTER 13

CONCLUSION AND IMPLICATIONS

This paper analyzes activities performed and problems encountered in each of the processes undertaken in the case. Information was collected through interviews, participation and observation of activities undergone. The whole system development and implementation process is broken down in ten phases : information analysis, system design, development, testing, system installation, training, system launch, review, system maintenance, and operation management. Concrete examples and problems are used to illustrate various abstract principles and techniques. The information is then generalized to give guidelines to implementation of new technology.

In information analysis phase, which involves definition of business objectives, examining current system and proposing system solutions, users' involvement is crucial as they are an important source of information for analyzing information and processing requirements of the system. The mutual cooperation and communication between users and system developers should be encouraged. Besides users' involvement, documentation is another source of information, thus the importance of documentation needs to be emphasized at all levels. And, documentation should be done jointly by users and system developers so that the information contained will be more complete, with both human and technical aspects.

Different alternatives to serve the business objectives should be explored so that the best alternative can be identified. Various techniques such as user groups and prototyping can be used to help users to propose system solutions. Actual users who will operate the new system should be involved in proposing system solutions in order to make it operable in working environment and more acceptable to users who use it. Overviews and guidelines should be given to users in defining their requirements such that the requirements will be more comprehensive and realistic. Standard procedures should also be set for user to specify their requirements. Also, actual users performing daily operations of the system and those operating the new equipment in outlets should be involved in system definition.

Since users and system developers are generally under different units with different working environment and management, it is very easy to fall in the trap of a “We-They” relationship during system definition. Some steps can be used to reduce the "We-They" syndrome : establishing a joint participation cooperative base; securing senior management commitment from a level above all units; cross fertilizing the environment by putting people from each unit into the other for long term project; using or building a common language to avoid getting trapped into communication voids; setting up open dialogues between all unit managers and keep them informed.

In system design phase, business specification is translated into actual computer and manual processes. Feasibility assessment in various aspects such as legal feasibility, social feasibility, technical feasibility and operational feasibility should be performed. Old system, if present, should be considered in system design.

Functional specification, the major deliverable in this phase, must provide user with a clear view of the outcome of the systems product, the changes to the user organization, change in business process in accommodation of the changed system, risks of the application, and steps to be taken to make the system work. User participation will be helpful to ensure the functional specification contain operational issues.

During system development, two approaches, the top-down approach and the bottom-up approach can be used. System developers need to decide which approach to take. The top-down approach is feasible only when hardware and software are available at early stage and the system can be launched in phases. However, system developers should be aware of the impact of incremental change in users' operation when the system is launched incrementally. On the other hand, bottom-up approach is used when related hardware or software is available only at a late stage of the whole system development. Good communication between developers of different subsystems is crucial to ensure system integrity and compatibility between subsystems.

Briefing sessions should be conducted during the system development phase to discuss the features of the design so that other developers will have an overall picture of the system. This gives chance for sharing knowledge on the system and skills, as well as the chance to find errors and omissions. This sharing of knowledge also makes the whole development team more capable of maintenance work of the system. It provides visibility, highlighting and detailing of the project contents, deliverables and outcomes to all involved. Moreover, this creates wider awareness of the system and formally transfer ownership of a system from few developers to the whole project team, making the development effort less personal and more a group effort.

The problems encountered in each phase of technology implementation and the corresponding solutions are summarized in appendix A. This gives guidelines to implementation of new technology.

APPENDIX A

Summary on Implementation Problems

Task	Problems	Solution
Identification of Solution	Solution other than best alternative being used	Explore different alternatives before committing to a solution
Definition of system's scope	Scope of project expanded unnecessarily, causing cost over-running	Perform cost benefit analysis with adherence to the business objectives defined
Examining the current system	Incomplete or even no document available	• Use other sources of information like interviewing, questionnairing • Build a documentation library • Set guidelines and standards for documentation • Emphasis the importance of documentation
Examining the current system	Users being unwilling or unable to provide information	• Emphasis the importance of communication between users and system developers • Budget time and effort for users to provide information
Proposing system solution	Users find difficulties in defining requirements	Use other similar systems as source of ideas
Proposing system solution	User being unable to express their requirements precisely	Use experimental prototype for users to try alternative designs
Proposing system solution	User cannot understand features of new system	User explanatory prototype for users to visualize new features
Proposing system solution	Conflicts in requirements of different user groups	Conduct user group sessions to resolve conflicts
Proposing system solution	Requirements submitted being incomplete and misleading	Set guidelines and standard procedures for users to specify their requirements
Proposing system solution	Requirements fail to fit in actual operation	Involve daily operation users in defining the system
Proposing system solution	"We-They" syndrome exhibited between users and system developers	• Establish a joint participation cooperative base • Secure senior management commitment from a level above all units
System design	the designed system cannot meet legal or not operarble in working environment	Conduct feasibility studies not only in technical aspects in system design

Task	Problems	Solution
System design	Users feel discontent and surprised because some functions in old system missing or changed totally	Preserve old system features and achieve functional equivalence in system design
System design	Functional specification not userful as only technical details covered	Involve users in preparation of functional specification to include business and operation issues
System development	User being skeptical to any drastic change	Use top-down approach to build basic structure and add change incrementally
System development	User change requirements during system development	Frozen requirements and defer the extra requirements to next phase
System development	Development being personal rather than group effort	Conduct briefing sessions on new system to share knowledge and find errors and omissions
System development	System developers do not have the necessary skills	Budget time and effort to acquire the needed skills
System development	System developed has low quality	Address quality issues continuously during system development
Testing	Users do not know how to write test plan	• Set guidelines for writing test plan • Assign those involved in defining the system to write test plan
Testing	Users do not have time to do testing	• Budget time and effort for testing • Make testing part of user's job responsibility
Conversion	Incorrect data being brought to the new system	• Check validity of data before conversion • Perform checking on converted data
System launch	Disruption arises during system launch	Construct implementation plan and make it well understood to all involved parties
System maintenance	System developers cannot make good capacity planning	Users should provide information on projection of business volume

BIBLIOGRAPHY

Allen, Thomas J. Managing the Flow of Technology : Technology Transfer and the Dissemination of Technological Information within the R&D Organization. London, England: The MIT Press and London, England, 1977.

Andriole, Stephen J. Managing Systems Requirements : Methods, Tools, and Cases. New York: McGraw Hill Company, c1996.

Avgerou, Chrisanthi, and Cornford, Tony, Department of Information Systems London School of Economics. Developing Information Systems Concepts, Issues and Practice. Basingstoke: The Macmillan Press Ltd., 1993.

Connor, Denis. Information System Specification and Design Road Map. Englewood Cliffs, New Jersey: Prentice-Hall Inc., c1985.

Davidson, Harold F., Cetron, Marvin J., and Glodhar, Joel D. Technology Transfer. Leiden, The Netherlands: Noordhoff International Publishing, 1974.

Kozar, Kenneth A. (University of Colorado/Boulder, College of Business). Humanized Information Systems Analysis and Design. New York: McGraw Hill Book Co., c1989.

Lucas, Henry C. JR. The Analysis, Design, and Implementation of Information Systems. New York: McGraw-Hill, Inc., c1976.

Samet, P.A. Information System Methodologies. New York: Wiley Heyden Ltd. on behalf of the British Computer Society, c1983.

Shelly, Gary B. Business Systems Analysis and Design. Fullerton, CA: Anaheim Publishing Co., 1975.

Stair, Ralph M. Principles of Information Systems A Managerial Approach. Danvers, Mass: Boyd & Fraser Publishing Company., c1996.

Prepared by National Science Foundation. Technology Transfer and Utilization - Recommendations for Redirecting the Emphasis and Correcting the Imbalance. Washington, D.C.: National Academy of Engineering, Feb 1974.